DebConf17 Report

Kazuhiro Hayashi, Toshiba Corporation
Japan Technical Jamboree 62
Sep 1st, 2017

About DebConf17

- **Dates**
  - 6th Aug. - 12th Aug. 2017 (Held every summer)
- **Venue**
  - Collège de Maisonneuve, Montreal, Canada
- **Official website**
  - https://debconf17.debconf.org/
- **Attendees**
  - DPL (Debian Project Leader): Chris Lamb
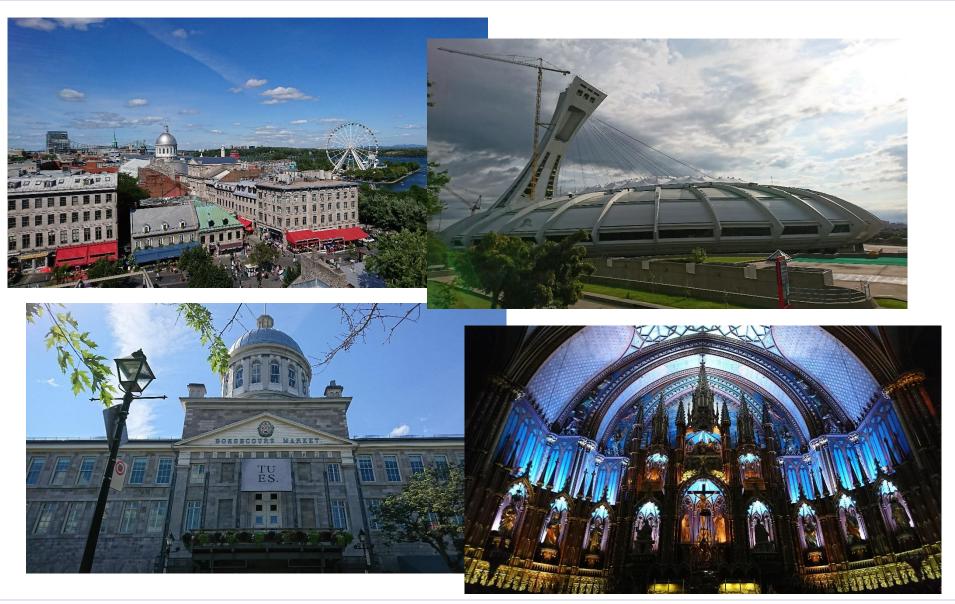 - Debian developers
  - Users, Sponsors, …

Venue

Montreal

About DebConf17

- **Topics**
  - Over 150 talks and BoFs related to Debian
    - Discussions about future development of Debian
    - Basic / advance tutorial of Debian tools
    - Debian derivatives, Debian-based custom OS
      - Derivatives: https://wiki.debian.org/Derivatives/Census, http://deriv.debian.net/
    - Infrastructure for building, testing Debian
    - Reproducible build
    - Security updates, LTS
    - License scanning
    - Container, Docker
    - Applying Debian to industrial products
  - Free discussions with attendees (Lunch, Dinner, Party, Free time…)

Building a PKI and PGP clean room

- **https://debconf17.debconf.org/talks/66/**
- **Debian Live based clean room for PGP/PKI**
  - https://wiki.debian.org/OpenPGP/CleanRoomLiveEnvironment
- **Purposes**
  - Make it easy (and secure) for everyone to setup GPG
    - To remember command sequences to setup GPG is sometimes hard for people new to the community
- **How it works**
  - Debian Live CD with networking disabled (= Clean room)
  - Keep the master key in the clean room
  - Use USB device to import/export public keys

Endless OS

- **https://debconf17.debconf.org/talks/41/**
- **Debian-based distro (x86-64 & ARMv7)**
  - Built from Debian binary packages on OpenBuildService
  - Filesystem is managed by OSTree, Apps are managed by Flatpak
- **OSTree**
  - Atomic upgrades & rollbacks for entire OS
  - Delta updates
  - Secure distribution through HTTPS & GPG signatures
- **Flatpak**
  - Applications are sandboxed and independent from others (OS)
- **Tools & Workflow**
  - https://github.com/dbnicholson/deb-ostree-builder
  - Check updates => Create OS tree from debootstrap + pkglist
    => Commit the new tree to OSTree repo
    => Publish the OSTree to run-time environment

Debian for Medical Software

- **https://debconf17.debconf.org/talks/165/**
- **Roche Linux**
  - Debian binary package based (wheezy/jessie)
    - Started from PTXdist (cross-build everything), but moved to Debian
  - Real-time kernel
  - Derivative?
- **Target**
  - Diagnostic instruments / laboratory
- **Evaluation criteria**
  - Cost efficiency (migration, license, commercial support, etc.)
  - Flexibility, usability, reliability
  - Long term support
  - Reproducibility of the build process

A distro for industrial R&D and engineering

- **https://debconf17.debconf.org/talks/145/**
- **Target**
  - Scientific computing: Modeling, Simulation, Visualization
  - Scientific workstation / laptop, High performance clusters, etc.
- **Scibian by EDF**
  - Debian derivative, don't fork, don't recompile
  - Customize for their needs
  - Define a general architecture for cluster
- **Additional features**
  - Add non-Debian software (ex: libtau)
  - Upgrade package version
  - Re-compile packages to improve performance, fix bug, etc.
    - Contribute Debian LTS first if possible
  - Backport H/W supports (Manufacture need specific version)
  - 6+ year support, Life cycle of OS/HW/App
    - Still maintaining squeeze by themselves

Apertis

- **https://debconf17.debconf.org/talks/146/**

- **Aperits**
  - Debian-derived platform for infotainment in automotive vehicles
  - Based mostly on Ubuntu + several Debian packages
  - AppArmor, OSTree/Flatpak, Wayland, Gstreamer

- **Infrastructure / Work flow**
  - Debian srcpkg + Other S/W (git) => OBS(build) => Binary packages => Image creator (Linaro image tools) => RFS => Target
  - Automation & Testing: Jenkins + LAVA
  - Maintains internal package repositories (one per project)
  - Image build: ospack(H/W independent) + hwpack (H/W dependent)
  - Testing: debian/rules test

Isar

- **https://debconf17.debconf.org/talks/147/**
- **Isar**
  - Image builder, not a distribution
  - Integration System for Automated Root filesystem generation
- **Requirements**
  - One command, on-demand building
  - Ready-to-use images
  - Reproducible results
  - Support for managing commodities (product lines)
  - Security updates, 10+ years maintenance
- **Uses**
  - Debian binary packages
  - bitbake & YoctoProject's Structure, layering, workflow

Extended LTS

- **Discussion with Debian LTS & companies using Debian**
- **Common situation**
  - 7-10 years support
  - As a liability, not a business
- **Topics**
  - Package repository for extended LTS
    - Debian 6: Archived, no further uploads are possible
    - Debian 7 or later: could be kept open by Debian project
    - Or use some shared infrastructure
  - Only support a small subset of Debian archive
  - Kernel version
    - CIP: 4.4, not part of Debian
  - How the efforts by the attending companies are recognized on www.debian.org
  - How the extended LTS could get executed between companies
  - ML: debian-companies, debian-lts-private, ML of CIP, others
- **Future plan**
  - Keep source packages in Debian main repository for 10 years at least
  - Discuss with companies who are developing (sponsoring) Debian
  - Discuss with the leader of Debian LTS project

Will there be Debian in your next BMW car?

- **https://debconf17.debconf.org/talks/33/**
- **In the car world**
  - More general hardware and software
  - Functionally close to regular PCs (and smartphones)
    - e.g. entertainment
  - Going Deeper
    - GenIVI, YP, OE, bitbake, GIT, Jenkins, Gerrit, systemd, wayland
- **Use Yocto Project / OE based system, but**
  - Basic packages need maintenance (patches, updates, dependencies)
  - Runtime dependenciescan trigger rebuilds
  - SDK can grow quite fast (become big?)
  - Cross-server build result chaching is extremely tricky
  - Builds are not reproducible
- **Where is Debian?**
  - Package the auto components for exposure
  - Perfect reproducible builds
  - Push changes upstream
  - etc.

Debian installer

- **https://debconf17.debconf.org/talks/167/**
- **https://debconf17.debconf.org/talks/116/**
- **Resent changes (stretch & buster)**
  - Full CD sets are gone
  - Support mips64el, powerpc
  - Graphic installer (default to X11)
  - HTTPS support (wget instead of busybox)
  - Busybox Maintenance
    - Included in CIP core packages
  - Testing: kadit (Compare screenshot)
- **Requirements for Debian installer**
  - Size reduction for small RAM
    - e.g. Reduce kernel size, remove graphical installer
    - Suggestion: Use the default (minimized) Debian rootfs
  - Reproducible build for debian-installer & debian-cd
  - Improving partman recipes
  - Remote access to installer (ssh)

Patterns for Testing Debian Packages

- **https://debconf17.debconf.org/talks/34/**
- **About**
  - "Pattern language" for automated software testing
  - Based on a paper: https://deb.li/pattestdeb
  - For autopkgtest / DEP8
- **Patterns**
  - Reuse existing tests
    - Upstream sometimes provides tests
  - Test the installed package
    - File paths
  - Clean and disposable test bed
    - Reproducibility
    - Use virtualization or container to provide fresh system
  - Acknowledge know failures
    - A test may fails for several reasons
    - If not fatal, add to "know failures list"
  - Automatically generate test metadata
    - Remove duplicated test definitions
    - autodep8
  - Smoke tests
    - Covers the main & basic functionality test of a system
    - ex: COMMOND --version
  - Record Interactive Session
    - Sometimes writing automated test are not so easy => record sample interactions

Towards Easier Security Patch Porting

- **https://debconf17.debconf.org/talks/166/**
- **Tool to make security patch porting easier**
  - https://github.com/PatchPorting
  - Ex: Automate backporting upstream security patches to old version
- **How it works**
  - Automate all processes
    - find patches, backports, build test, update web
  - If there are multiple solutions, provide multiple patches
  - Change strategy how to apply patch (default, or others)
- **Goal**
  - to have a platform where security patches are shared

Reproducible build

- **https://debconf17.debconf.org/talks/91/**
- **https://debconf17.debconf.org/talks/14/**
- **Definition**
  - https://reproducible-builds.org/docs/definition/
  - Cross distro contributions
- **Progress**
  - https://tests.reproducible-builds.org/debian/reproducible.html
- **How many packages can be reproducible now?**
  - with specific path settings, 94%
  - without unspecific paths, around 50%
- **.buildinfo**
  - Description of the build environment
  - built by developer or build
- **Causes of non-reproducibility**
  - specifics of the machine/person building the packages
  - time
  - randomness
  - Build path (Hard to fix)
  - DEB_BUILD_OPTIONS, DEB_BUILD_PROFILES
  - Versions of installed build dependencies
- **Source + .buildinfo => reproducible binaries**
- **See deb-buildinfo(5)**

Lessons learned

- **Many chances to find out common values with Debian**
  - Ex: Minimize the default Debian system
    - To apply the default Debian to various embedded systems
    - Could replace Debian installer rootfs?
- **Opportunities to understand Debian's opinions quickly**
  - Ex: debian-cross
    - Good place to discuss & contribute about cross-building
    - Actually 2 or 3 people active
    - Issue: Concrete results should be shared in community
  - For Debian developers, good chance to make progresses by discussing other developers without e-mail
- **A place where companies can collaborate**
  - Ex: Extended LTS, Security patch porting

DebConf18

- **https://wiki.debconf.org/wiki/DebConf18**
- **Venue**
  - National Chiao Tung University
  - Hsinchu City, Taiwan
- **Date**
  - Sun, July 29 – Sun, August 5
- **Become a sponsor!**
  - https://debconf17.debconf.org/sponsors/become-a-sponsor/